PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 11/14 — A1

(11) International Publication Number: WO 95/22794

(43) International Publication Date: 24 August 1995 (24.08.95)

(21) International Application Number: PCT/US95/01916

(22) International Filing Date: 16 February 1995 (16.02.95)

(30) Priority Data:
08/198,343 18 February 1994 (18.02.94) US

(71) Applicant *(for all designated States except US)*: APPLE COMPUTER, INC. [US/US]; One Infinite Loop, Cupertino, CA 95014 (US).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: YEN, John [US/US]; 438 Palma Avenue, El Granada, CA 94018 (US).

(74) Agent: PETERSON, James, W.; Burns, Doane, Swecker & Mathis, P.O. Box 1404, Alexandria, VA 22313-1404 (US).

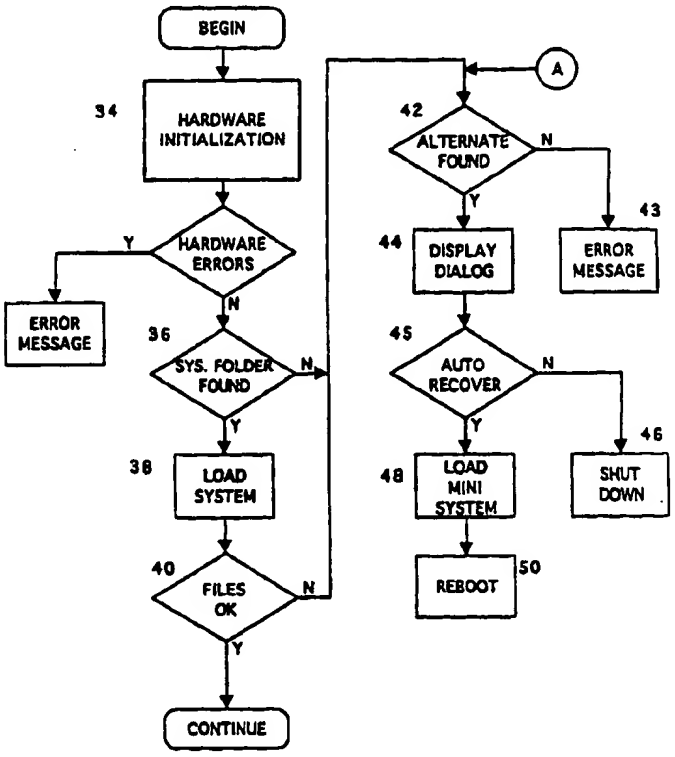
(81) Designated States: AM, AT, AU, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, JP, KE, KG, KP, KR, KZ, LK, LR, LT, LU, LV, MD, MG, MN, MW, MX, NL, NO, NZ, PL, PT, RO, RU, SD, SE, SI, SK, TJ, TT, UA, US, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), ARIPO patent (KE, MW, SD, SZ, UG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: SYSTEM FOR AUTOMATIC RECOVERY FROM SOFTWARE PROBLEMS THAT CAUSE COMPUTER FAILURE

(57) Abstract

A system for recovering from certain types of system software startup problems employs a user-hidden secondary startup volume stored in the computer. During a normal startup procedure, if an error is detected which would normally result in a startup failure, the computer's startup routine branches to an alternate startup application stored in the secondary volume. This startup application boots the computer from a minimal operating system stored in the secondary volume. As a result, the user is not left with a non-functioning computer. As further features of the invention, the startup application can attempt to automatically fix the detected problem, or it can suggest possible steps to be taken by the user, in order to fix the problem that resulted in the need to use the alternate startup application.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**GB**	United Kingdom	**MR**	Mauritania
AU	Australia	**GE**	Georgia	**MW**	Malawi
BB	Barbados	**GN**	Guinea	**NE**	Niger
BE	Belgium	**GR**	Greece	**NL**	Netherlands
BF	Burkina Faso	**HU**	Hungary	**NO**	Norway
BG	Bulgaria	**IE**	Ireland	**NZ**	New Zealand
BJ	Benin	**IT**	Italy	**PL**	Poland
BR	Brazil	**JP**	Japan	**PT**	Portugal
BY	Belarus	**KE**	Kenya	**RO**	Romania
CA	Canada	**KG**	Kyrgystan	**RU**	Russian Federation
CF	Central African Republic	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CG	Congo			**SE**	Sweden
CH	Switzerland	**KR**	Republic of Korea	**SI**	Slovenia
CI	Côte d'Ivoire	**KZ**	Kazakhstan	**SK**	Slovakia
CM	Cameroon	**LI**	Liechtenstein	**SN**	Senegal
CN	China	**LK**	Sri Lanka	**TD**	Chad
CS	Czechoslovakia	**LU**	Luxembourg	**TG**	Togo
CZ	Czech Republic	**LV**	Latvia	**TJ**	Tajikistan
DE	Germany	**MC**	Monaco	**TT**	Trinidad and Tobago
DK	Denmark	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**US**	United States of America
FI	Finland	**ML**	Mali	**UZ**	Uzbekistan
FR	France	**MN**	Mongolia	**VN**	Viet Nam
GA	Gabon				

SYSTEM FOR AUTOMATIC RECOVERY FROM SOFTWARE PROBLEMS THAT CAUSE COMPUTER FAILURE

Field of the Invention

The present invention is directed to detection and recovery from problems that cause operational failures in computers, and in one particular embodiment to a system that permits a computer to boot despite the presence of a software problem that could otherwise cause a startup failure.

Background of the Invention

When a computer is first started, it proceeds through certain startup stages which check for particular types of errors. If no errors are detected, the computer loads an operating system into its main memory and carries out other tasks associated with initial startup. If, however, an error is detected during the startup stages, further procedures are interrupted and the computer may display a warning to the user, to provide an indication of the detected error. For example, if an operating system file is not stored at an expected location, the computer may display the message "Unable To Find System Folder."

For sophisticated computer users, the steps necessary to correct for many types of errors that cause startup failures may be readily apparent. For example, if the system folder cannot be located, an experienced user may realize that an emergency disk containing the system folder needs to be located and inserted in a disk drive, or a file needs to be moved from one location to another. A novice user, however, may not understand the error

message, and therefore not know how to recover from the error condition. In either case, the user is faced with a computer that doesn't work, and may spend a significant amount of time and effort attempting to overcome the problem that caused the failure.

In the past, some efforts have been made to overcome certain types of errors that interfere with proper operation of the computer. For example, some computers are constructed with hardware fault-tolerant systems. In these types of systems, certain faults which might otherwise cause a failure can be overcome or circumvented. While this approach has utility in certain circumstances, it is not generally applicable to all types of computers. Rather, the fault tolerance must be specifically designed for each different type of computer. In addition, this approach does not offer the capability to correct software-related problems, which are the more common source of operational failures in general purpose computers.

Other approaches to this problem have employed software stored within the computer system that attempts to identify and fix certain types of file system errors. Some of these approaches are user-invoked, rather than being automatic, and therefore not likely to be used by novice users. In addition, they are specific to the particular system on which they are stored, rather than being generally applicable to all types of computers.

Accordingly, it is desirable to provide a system for recovering from software errors that has general applicability to any type of computer system and requires only minimal system resources.

Brief Statement of the Invention

In accordance with the present invention, the foregoing objective is achieved by means of a user-hidden secondary volume or partition in the computer's permanent storage mechanism, e.g., hard disk. If an error is detected which would normally result in an operational failure, the computer branches to recovery software stored in the secondary volume. For example, in the case of startup errors, the recovery software can include an alternate startup application which enables the computer to be booted. In one embodiment, the startup application installs a minimal operating system on the primary volume and then restarts the computer. Due to the presence of the minimal operating system installed in the primary volume, the computer is able to start. As a result, the user is not left with a non-functioning computer.

As further features of the invention, the recovery software can attempt to automatically fix the detected problem, or it can suggest possible steps to be taken by the user, in order to fix the problem that resulted in the initial failure.

Since the error detection and recovery system of the present invention is software-based, and does not rely upon specialized hardware, it has general applicability to any type of computer system. Furthermore, it requires only the use of minimal system resources to operate. It provides the user with an automatic recovery from software problems, rather than simply detecting a startup problem and informing the user. Thus, the user does not need to have

any familiarity with the operation of a computer in order to overcome problems, and does not have to pursue manual procedures to fix a problem in order to make the computer operate.

Further features and advantages of the invention are explained hereinafter with reference to preferred embodiments illustrated in the accompanying drawings.

Brief Description of the Drawings

Figure 1 is a general block diagram of the major components of a computer system in which the present invention can be employed;

Figure 2 is a flowchart of the general operation of a software-error recovery system in accordance with the present invention;

Figure 3 is a diagram of a dialog screen display that appears when a software-error is detected;

Figures 4-7 are diagrams of screen displays that appear when the user has chosen to boot the computer with the alternate startup application;

Figure 8 is a flowchart of an alternate procedure for invoking the recovery system; and

Figure 9 is a flowchart of a procedure for monitoring startup to detect a reason for failure.

Detailed Description

In order to facilitate an understanding of the principles and features of the present invention, it is described hereinafter with reference to its implementation in a personal computer. In certain instances, specific reference is made to features of a Macintosh brand computer, supplied by Apple Computer, Inc., for ease of understanding. It will be appreciated, however, that the applicability of the invention is not limited to this particular embodiment. Rather, the invention can find utility in any type of general purpose computer in which it is desirable to overcome software-related problems.

The primary components of a general purpose computer, in which the present invention can be implemented, are illustrated in block diagram form in Figure 1. The structure of the computer itself does not form part of the present invention. It is briefly described here for subsequent understanding of the manner in which the features of the invention cooperate with the structure of the computer.

The three main components of a general purpose computer comprise a processor, a user interface for data entry and output, and a data storage or transfer device. Referring to Figure 1, the computer 10 includes a central processing unit 12. Additional processors can also be included in the computer. The data storage device 14 includes a main memory which is typically implemented in the form of a random access memory 16. The data storage can also include a static memory such as a read only memory 18, and

other permanent storage, such as a magnetic or optical disk 20. The data entry portion of the user interface 22 can be a keyboard 24 and/or a pointing or cursor control device 26 such as a mouse, trackball, pen or the like. The data output portion typically includes a display device 28, such as a CRT monitor or an LCD screen. The CPU 14 communicates with each of these devices through a bus 30.

A computer of the general type illustrated in Figure 1 can be embodied in several different forms, including a mainframe computer, a network terminal, a personal computer, and a palmtop computer, or personal digital assistant (PDA).

When the computer is first started, it runs a startup routine that is stored on its internal ROM 18. As part of this startup routine, the computer proceeds through various stages to determine its configuration, e.g., the number and addresses of the various disk drives, and whether its hardware components are properly functioning. For example, it may check the status of communications with the various input and output devices, such as the keyboard 24 and the cursor control device 26. In addition, it can perform a check of the RAM 16 to be sure it is properly accessible.

If no errors were detected in the hardware, the startup procedure then loads an operating system into the RAM 16. Typically, at least part of the operating system is stored in permanent storage, such as the ROM 18 and/or the disk 20. The permanent storage may be divided into a number of volumes or partitions. The startup routine expects to find required parts of

the operating system in a particular volume, referred to herein as the main or primary volume. If the necessary operating system components are found in the main volume, the integrity of certain system files are checked. If no corruption is found in these files, the operating system is loaded into the RAM 16, and the CPU 14 then operates in accordance with the instructions provided by the operating system.

If, however, the operating system file cannot be located in the main volume, or if certain files are found to be corrupted, a startup failure occurs. In the past, when this situation occurred, an error message was presented on the display, to inform the user that startup could not be completed due to a detected problem. If the user was sufficiently familiar with the operation of the computer to correct the problem, appropriate actions could be carried out. For example, a startup disk which contained the proper operating system files could be inserted in a disk drive, and the system restarted. If corrupted files were found during the initial startup procedure, utility software could be employed to attempt to repair the files. However, these procedures required the user's input and time to carry out the appropriate actions. Furthermore, a novice user, particularly one who had just purchased the computer, is not likely to understand the nature of the detected problem, and therefore how to correct it.

To overcome these difficulties, in accordance with the present invention an alternate startup application is stored on a secondary or recovery volume of the disk or other permanent storage. If certain types of problems which

prevent a normal startup are detected, the alternate startup application in the recovery volume is employed to boot the computer. For example, the startup application can install a minimal bootable system on the primary volume, and then restarts the computer with this minimal system. In the context of a Macintosh computer, for example, the minimal bootable system can comprise a system folder, a user interface application, such as the Finder, and one or more system enablers, e.g., system.init files. Due to the presence of the minimal bootable system in the main volume, the computer is able to start. As a result, the user is not faced with a non-functioning computer.

Alternatively, the startup application can directly boot the computer from the minimal system in the recovery volume, and thereby avoid the need to force a restart of the computer.

The recovery software is located in a separate area of permanent storage, rather than the main area, to ensure its reliability. In the context of the present invention, that separate area could be a distinct partition on a disk, or it could be a separate volume, i.e. file system, on a partition that contains other volumes. Further in this regard, the secondary volume or partition need not be located on the same storage device as the main volume from which the system software is normally loaded. It could be located on a different device, such as flash memory or a network server. The only requirement is that the secondary volume be accessible during the startup process before the main volume is read.

The use of a separate volume for storing the recovery software is based upon certain features which ensure its reliability. Because it is separate, the recovery volume is rarely, if ever, written to during normal system operation, and therefore relatively safe from damage. In this regard, it is preferable that the recovery volume be hidden from the user, to further minimize the likelihood of inadvertent damage. When access does occur, it is either automatic or it takes place only in specified circumstances, e.g. a user response to a specific prompt. As such, all access to the recovery volume is under controlled conditions.

The overall operation of the recovery system of the present invention is explained in greater detail hereinafter with reference to one embodiment depicted in the flow chart of Figure 2 and various user interface screens illustrated in Figures 3-7. More detailed features of the operation are then explained with reference to the flow charts of Figures 8 and 9. Referring to Figure 2, when the computer system is first started, or booted up, it undergoes an initialization routine 34 to determine its configuration and detect hardware errors. If no errors are detected, the system proceeds to load the operating system into the main memory 16. As a first step in this process, the computer determines whether a system folder is located in the main volume of the disk (Step 36). If so, appropriate files from the folder are loaded into the main memory (Step 38), and checked to determine whether they are corrupt (Step 40). If the files are successfully loaded and found to have no corruptions, the computer continues with the remainder of

the booting process, under the control of the operating system.

If the operating system cannot be located in the main volume, or if one of the system files is found to be corrupted, the computer proceeds to Step 42, to determine whether an alternate startup application is located in a secondary volume. If no alternate startup application is present, an error message 43 is displayed, in a manner similar to the operation of a conventional computer when system software errors are encountered.

If the alternate startup application is present in the secondary volume, a recovery application dialog box, as illustrated in Figure 3, is displayed to the user (Step 44). This dialog box informs the user that a system software problem was encountered during the startup procedure, and provides the user with two options for proceeding. If the user is capable of fixing the problem, he can choose to shut down the computer, at Step 45, and carry out the necessary recovery steps. In response to selection of this option, the computer automatically shuts down at Step 46, to enable the user to carry out the appropriate procedures. If, however, the user is not sufficiently familiar with the operation of the computer to undertake such an endeavor, or simply chooses not to do so, he can allow the alternate startup application to boot the computer from a minimal system, and thereby overcome the detected problem.

If the user decides to allow the alternate startup application to proceed, the computer loads the minimum bootable system into the main volume of the disk at Step 48. For example, the contents of the recovery volume,

except the startup application, can be copied to the main volume. The startup application then reboots the computer at Step 50. With the minimal system installed on the main volume, the problem detected at Step 36 or Step 40 is no longer present, and the computer is able to complete the boot-up process. Preferably, the software which is represented by the flow chart of Figure 2 is present as program code in the hard disk driver that is run during startup.

Alternatively, instead of copying the minimal system to the main volume and forcing a restart of the computer at Step 50, the startup application can proceed to directly boot the computer from the system in the recovery volume. In this case, the operation of the startup application can be made totally transparent to the user.

When the booting process is completed with the use of the alternate system, an appropriate form of notification can be provided to the user. For example, a different desktop display might be presented, to indicate that something other than the normal system is being used. Alternatively, or in addition, the user can be presented one or more informational screens which provide guidance on fixing the problem that was initially encountered. An example of suitable informational screens is illustrated in Figures 4-7. Referring to Figure 4, the initial informational screen can identify possible reasons for the problem encountered during startup. Subsequent screens, such as those shown in Figures 5-7, can guide the user through appropriate steps for attempting to correct the problem.

In some situations, the failure to properly boot the computer may not be due to an operating system error. For example, power to the computer might be interrupted during the startup procedure, prior to the time that the operating system is loaded into the main memory. This power interruption may result in an error indication. However, in this situation it is not necessary that the recovery mechanism be accessed as a result of the error. Therefore, rather than invoking the recovery mechanism after any incomplete boot, it is preferable to allow one incomplete boot without a detected error, and then invoke the recovery mechanism if there is another such incomplete boot.

A flowchart which depicts the operation of the system in accordance with this preferred embodiment is illustrated in Figure 8. Referring thereto, during initialization (Step 34 of Figure 2), an index or counter, i, can be incremented to indicate that a boot process has been initiated (Step 52). If the computer is successfully started, this index can be reset to zero. If, however, the booting of the computer is interrupted, the counter will remain at a value greater than zero. At Step 54, the computer determines whether the value of the index is greater than one. If so, at least one incomplete boot has previously occurred. At this point, the computer determines, at Step 56, whether an operating system error was recorded during the previous incomplete boot. If so, the process proceeds to Step 42 of Figure 2, to implement the recovery mechanism.

If no operating system error was recorded during the previous incomplete boot, the system determines whether the index has a value greater than two, at Step 58. If so, at least two incomplete boots have previously occurred. Therefore, the recovery mechanism is again invoked to enable the computer to be booted. Otherwise, the normal startup procedure continues.

In the embodiments of the invention depicted in the flowcharts of Figures 2 and 8, the recovery system of the present invention automatically boots the computer from the minimal operating system installed on the recovery volume, if the user chooses that approach. As a further feature, it is possible to have the recovery system first attempt to fix the problem that resulted in the boot failure, rather than automatically boot from the minimal system on the secondary volume. For example, a novice user may delete necessary system files, or move them to an area where they cannot be normally accessed during startup. This situation would result in a negative response at Step 36 of Figure 2. Before booting from the alternate system in the recovery volume, it is possible to search the hard disks on the system to determine whether the necessary files are present. If they are found, they can be copied to the proper location on the disk, so that the normal booting process can continue.

An alternate type of fix might be to boot from a different disk, and notify the user. If such fixes can be implemented, it is not necessary to boot from the minimal operating system stored on the recovery volume.

In the operation depicted in the flowchart of Figure 2, the automatic recovery procedure is invoked if files intended for use at startup are missing or corrupt. These are not the only situations in which the automatic recovery system can be employed. It can also be invoked when other types of file system problems are encountered, and when files are being run or loaded at startup which should not be, such as viruses or conflicting system software. In the case of a virus, for example, the recovery system can implement a correction by detecting and fixing or replacing damaged files.

As a further feature, the recovery system can provide the user with various levels of information regarding the cause of the startup failure that caused the recovery system to be invoked. For example, in the procedure of Figure 8, if no operating system error was recorded after one incomplete boot, i.e. a negative response at Step 56, the booting process can be monitored to identify the location in the process at which the failure occurred. Such a procedure is illustrated in the flowchart of Figure 9. Referring thereto, if a negative response is received at Step 56, software patches are installed at Step 60, to monitor the next startup. The function of each patch is to monitor and record the completion of a startup procedure with which it is associated. Thus, as each procedure is completed, a log is generated which records the completed events (Step 62). The number of patches that are installed is determined by the amount of information that is desired to be obtained. For example, if only general information is needed, one patch could be installed for each major stage in the startup procedure. If

more detailed information is desired, a patch can be installed for each step or task to be performed during the startup procedure.

If another startup failure occurs after the monitoring patches have been installed, the last recorded task or stage in the log provides an indication of the stage or task at which the failure occurred. When the computer is booted by means of the alternate startup application in the recovery volume, a flag can be set to indicate this fact. After a successful boot, the startup application can check the state of this flag and, if it is set, read the log to inform the user of the location in the startup procedure at which the failure occurred, and the possible reasons therefore.

From the foregoing, it can be seen that the present invention provides a fault recovery system that automatically enables a software problem to be overcome, and thereby avoid the frustration of a non-working computer and the hassles associated with fixing it. For more sophisticated users who may be readily aware of the procedures necessary to correct software types of problems, the recovery system provides them with the option to carry out the necessary corrections themselves, rather than use the automatic recovery procedures. The system has very little impact on the computer's resources, since it requires very little extra processing during normal operation. More particularly, if a working system folder is present in the main volume, and no startup errors are encountered, the additional drive access that is required during startup, as a result of the procedures carried out in accordance with the present invention, is anticipated to be on the order of one second or less.

In the foregoing specification and accompanying drawings, the present invention has been described with respect to one particular implementation. It will be appreciated, however, that it can be embodied in other specific forms without departing from the spirit or essential characteristics thereof. The presently disclosed embodiments are therefore considered in all respects to be illustrative and not restrictive. The scope of the invention is indicated by the appended claims rather than the foregoing description, and all changes which come within the meaning and range of equivalency of the claims are intended to be embraced therein.

WHAT IS CLAIMED IS:

1. A method for recovering from problems associated with software that prevent normal operation of a computer, comprising the steps of:

establishing a secondary area in a storage mechanism of the computer which is separate from a main area that is normally accessed to obtain software files employed during operation of the computer, and storing in said secondary area at least those portions of software files which are necessary to perform a desired operation with the computer;

detecting a software-related problem that prevents normal operation of the computer;

loading the portions of the software files stored in said secondary area into said main area upon detection of said problem; and

operating the computer with the files loaded from the secondary area into the main area.

2. A method for recovering from problems associated with software that prevent normal operation of a computer, comprising the steps of:

establishing a secondary area in a storage mechanism of the computer which is separate from a main area that is normally accessed to obtain software files employed during operation of the computer, and storing in said secondary area at least those portions of software files which are

necessary to perform a desired operation with the computer;

detecting a software-related problem that prevents normal operation of the computer;

attempting to automatically fix a detected problem; and

operating the computer with the software files stored in said secondary area when said detected problem is not automatically fixed.

3. The method of claim 2 wherein said operating step comprises the steps of loading said portions of the operating system stored in said secondary area into said main area upon detection of said problem, and rebooting the computer to enable it to be started with the operating system loaded from the secondary area into the main area.

4. The method of claims 1 or 2 further including the steps of:

initiating a first startup of the computer;

detecting a first failure of the computer to start;

initiating a second startup of the computer;

monitoring tasks performed during said second startup of the computer;

detecting a second failure of the computer to start;

booting the computer from software files stored in the secondary area; and

indicating a problem associated with a task that was not performed during the second startup.

5. The method of claims 1 or 2 wherein said problem is the absence of operating system files in said main area of the storage mechanism.

6. The method of claims 1 or 2 wherein said problem is corruption of an operating system file stored in said main area of the storage mechanism.

7. The method of claims 1 or 2 wherein said problem is the detection of a file which should not be loaded during startup.

8. The method of claims 1 or 2 further including the step of providing a computer user with an option to shut down the computer instead of having said software files loaded into the main area.

9. The method of claims 1 or 2 wherein said secondary area is normally hidden from user access.

10. The method of claims 1 or 2 wherein said problem is a failure of the computer to properly boot upon startup, and said software files stored in the secondary area comprise a minimum bootable operating system.

11. A method for recovering from a problem that prevents normal startup of a computer, comprising the steps of:

establishing a secondary area in a storage mechanism of the computer which is separate from a main area that is normally accessed to obtain operating system files during startup procedures, and storing in said secondary area at least those portions of an operating system which are necessary to start the computer;

determining whether software necessary to boot the computer is stored in said main area;

attempting to locate said software if it is not stored in said main area;

copying the software to said main area if it is located and booting the computer from said copied software; and

booting the computer from said portions of the operating system stored in the secondary area if said software cannot be located.

12. In a computer, a system for recovering from software problems that interfere with proper startup of the computer, comprising:

a storage mechanism having at least a main storage area from which software is normally retrieved during startup of the computer, and a secondary area storing at least those portions of an operating system which are necessary to start the computer;

means for detecting a software problem that interferes with

proper startup of the computer;

means responsive to the detection of said problem for loading the portions of the operating system stored in said secondary area into said main area; and

means for rebooting the computer after said portions of the operating system have been loaded into said main area, to thereby enable the computer to be started.

13. In a computer, a system for recovering from software problems that interfere with proper startup of the computer, comprising:

a storage mechanism having at least a main storage area from which software is normally retrieved during startup of the computer, and a secondary area storing at least those portions of an operating system which are necessary to start the computer;

means for detecting a software problem that interferes with proper startup of the computer;

means for attempting to fix a detected software problem; and

means responsive to the detection of said problem for booting the computer from the portions of the operating system stored in said secondary area when the problem is not fixed by said attempting means, to thereby enable the computer to be started.

14. The system of claims 12 or 13 wherein said problem is the absence of operating system files in said main area of the storage mechanism.

15. The system of claims 12 or 13 wherein said problem is corruption of an operating system file stored in said main area of the storage mechanism.

16. The system of claims 12 or 13 wherein said problem is the detection of a file which should not be loaded during startup.

17. The system of claims 12 or 13 wherein said secondary area is normally hidden from user access.

18. The system of claims 12 or 13 wherein said secondary area is normally hidden from user access.

19. The system of claims 12 or 13 further including means for indicating the problem to a user of the computer.

20. The system of claim 19 wherein said indicating means includes means for recording events that occur during a failed startup, and means for identifying an event which was not successfully completed during the failed startup.

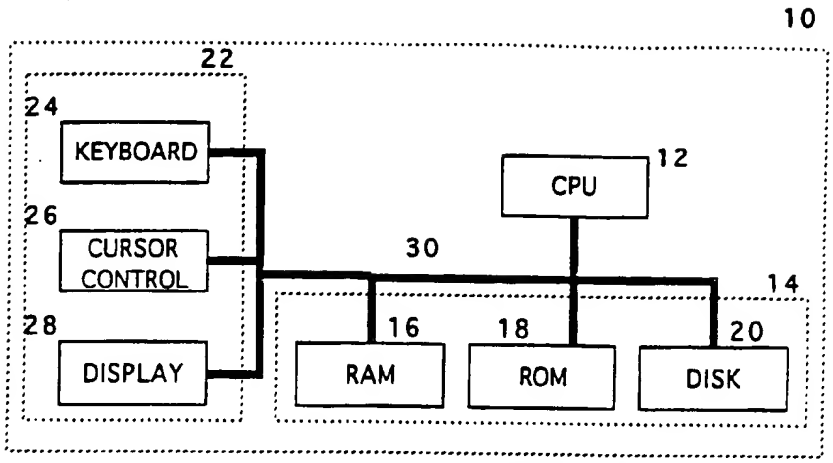

10
22
24
KEYBOARD
26
CURSOR CONTROL
28
DISPLAY
CPU
12
30
14
16
RAM
18
ROM
20
DISK

FIG. 1

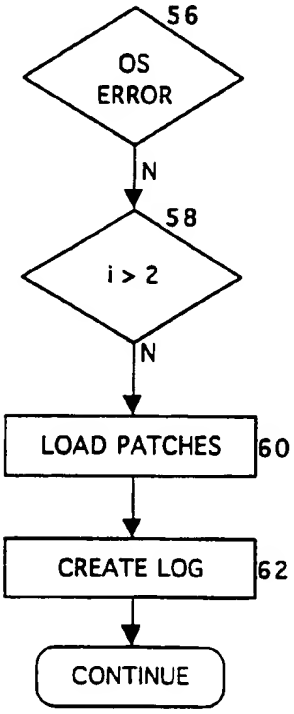

56
OS ERROR
N
58
i > 2
N
LOAD PATCHES
60
CREATE LOG
62
CONTINUE

FIG. 9

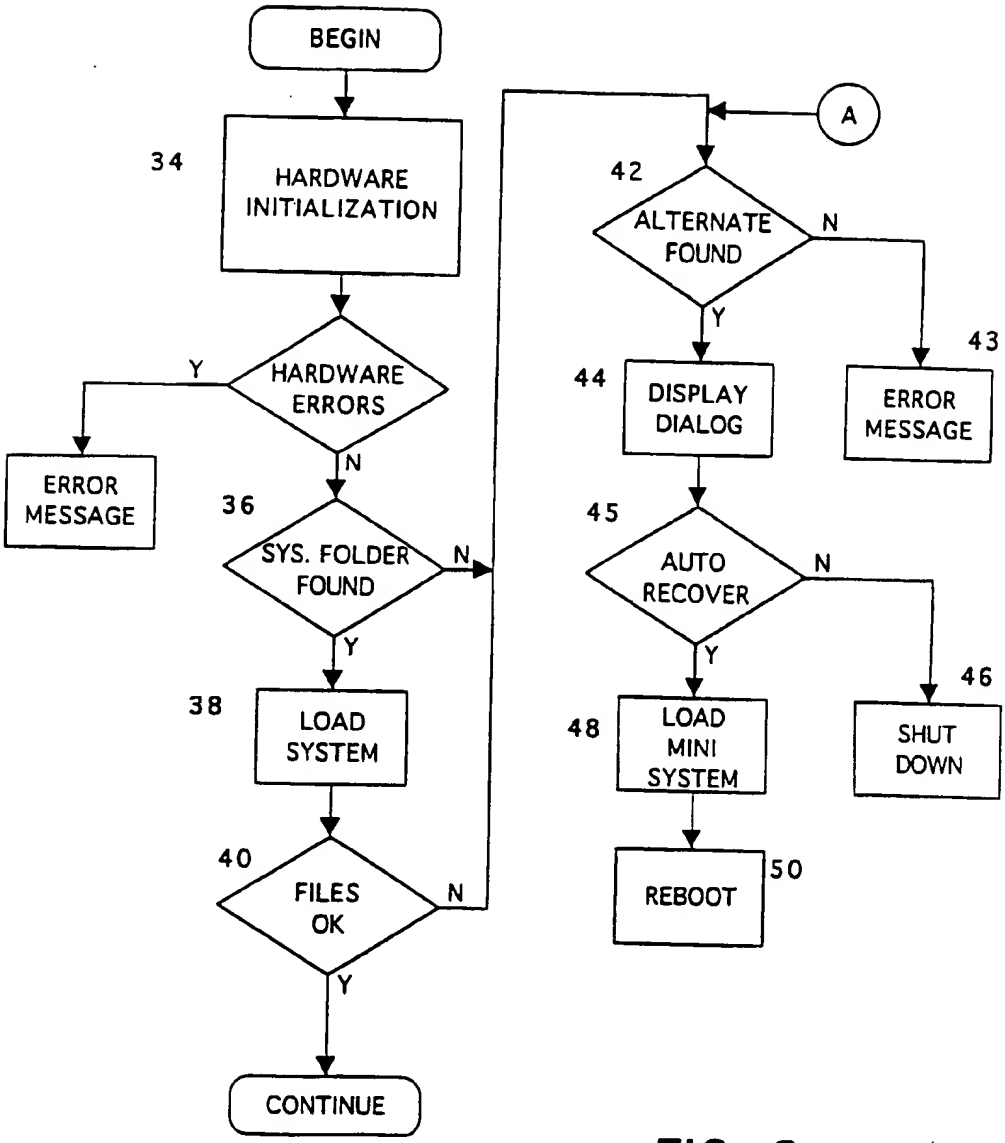

BEGIN
34
HARDWARE INITIALIZATION
HARDWARE ERRORS
Y
ERROR MESSAGE
N
36
SYS. FOLDER FOUND
N
Y
38
LOAD SYSTEM
40
FILES OK
N
Y
CONTINUE
A
42
ALTERNATE FOUND
N
Y
43
ERROR MESSAGE
44
DISPLAY DIALOG
45
AUTO RECOVER
N
Y
46
SHUT DOWN
48
LOAD MINI SYSTEM
50
REBOOT

FIG. 2

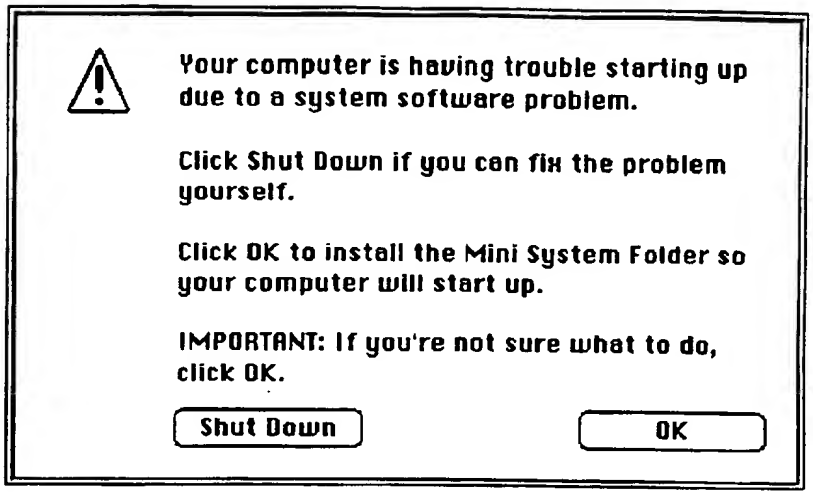

Your computer is having trouble starting up due to a system software problem.
Click Shut Down if you can fix the problem yourself.
Click OK to install the Mini System Folder so your computer will start up.
IMPORTANT: If you're not sure what to do, click OK.
Shut Down
OK

FIG. 3

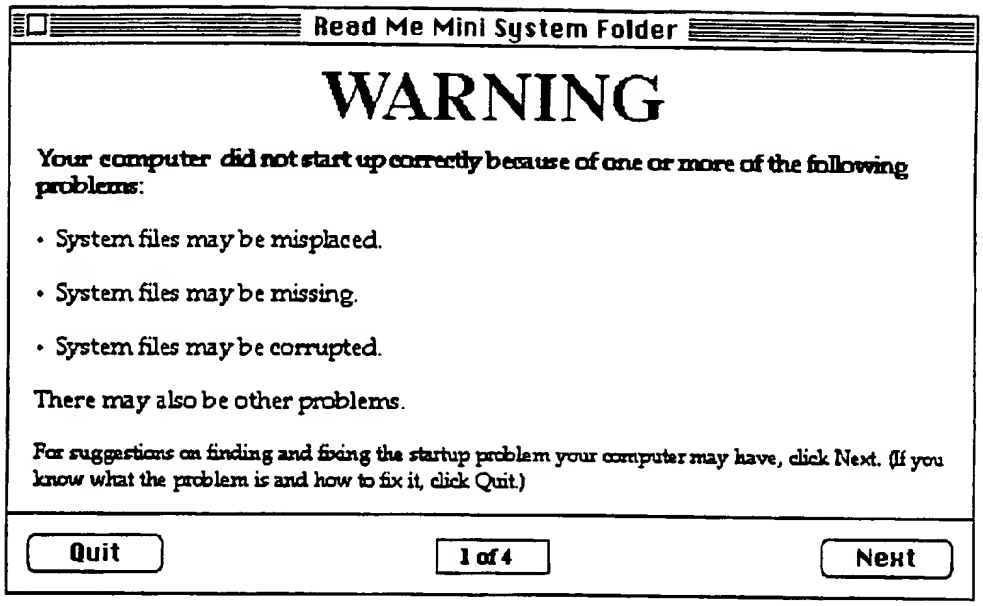

Read Me Mini System Folder
WARNING
Your computer did not start up correctly because of one or more of the following problems:
• System files may be misplaced.
• System files may be missing.
• System files may be corrupted.
There may also be other problems.
For suggestions on finding and fixing the startup problem your computer may have, click Next. (If you know what the problem is and how to fix it, click Quit.)
Quit
1 of 4
Next

Fig. 4

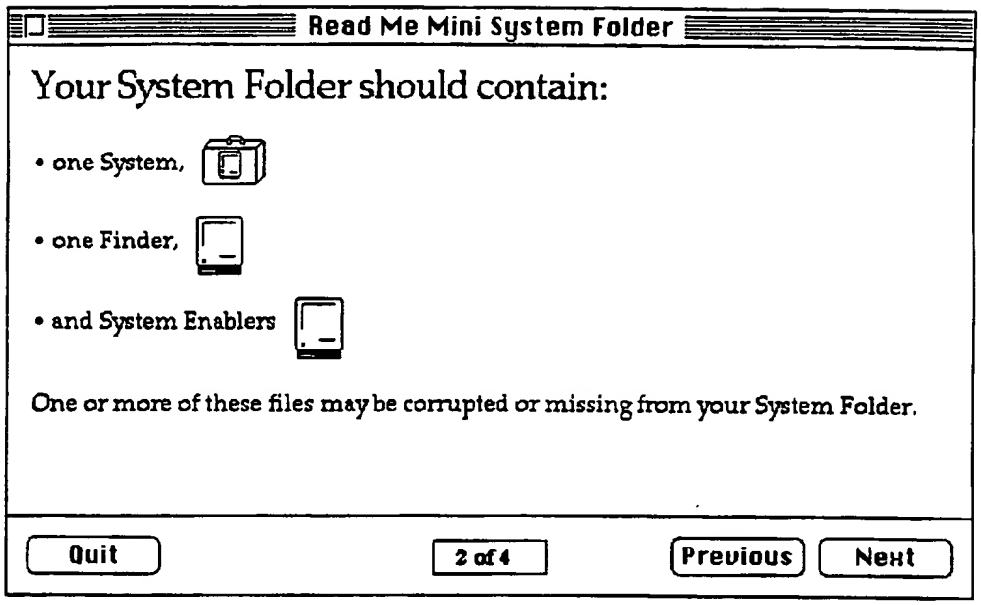

Read Me Mini System Folder
Your System Folder should contain:
• one System,
• one Finder,
• and System Enablers
One or more of these files may be corrupted or missing from your System Folder.
Quit
2 of 4
Previous
Next

FIG. 5

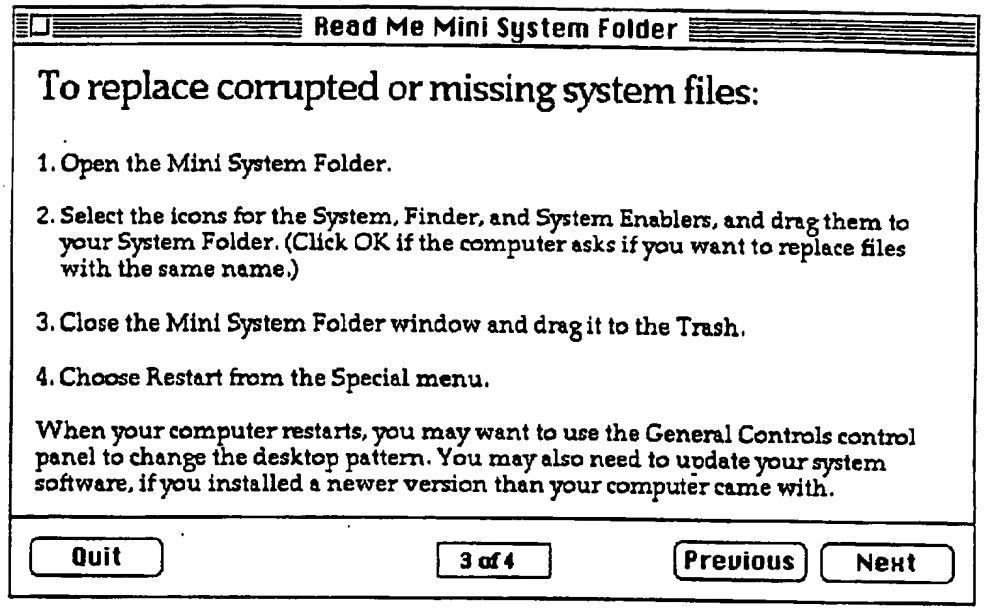

Read Me Mini System Folder
To replace corrupted or missing system files:
1. Open the Mini System Folder.
2. Select the icons for the System, Finder, and System Enablers, and drag them to your System Folder. (Click OK if the computer asks if you want to replace files with the same name.)
3. Close the Mini System Folder window and drag it to the Trash.
4. Choose Restart from the Special menu.
When your computer restarts, you may want to use the General Controls control panel to change the desktop pattern. You may also need to update your system software, if you installed a newer version than your computer came with.
Quit
3 of 4
Previous
Next

FIG. 6

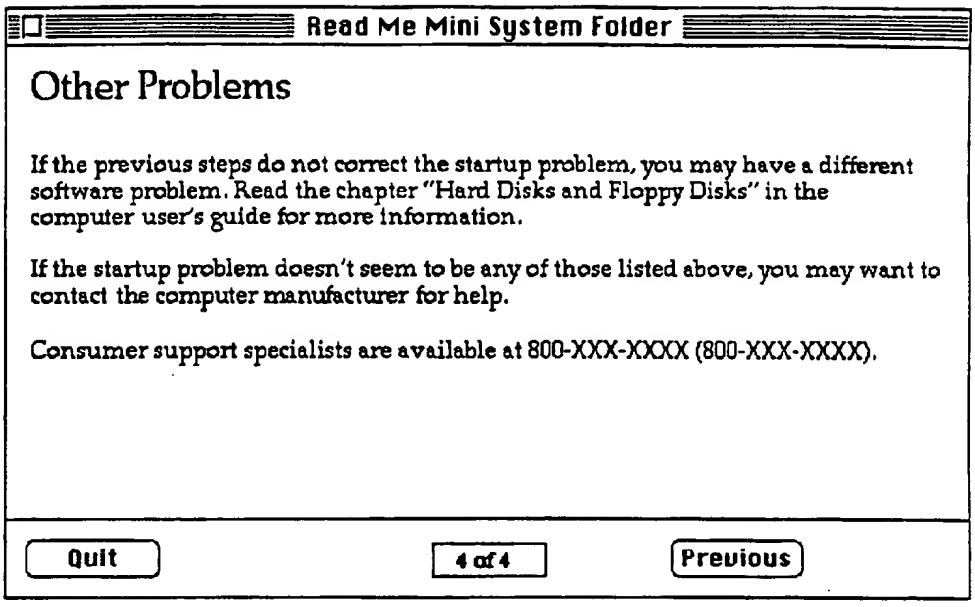

Read Me Mini System Folder
Other Problems
If the previous steps do not correct the startup problem, you may have a different software problem. Read the chapter "Hard Disks and Floppy Disks" in the computer user's guide for more information.
If the startup problem doesn't seem to be any of those listed above, you may want to contact the computer manufacturer for help.
Consumer support specialists are available at 800-XXX-XXXX (800-XXX-XXXX).
Quit
4 of 4
Previous

FIG. 7

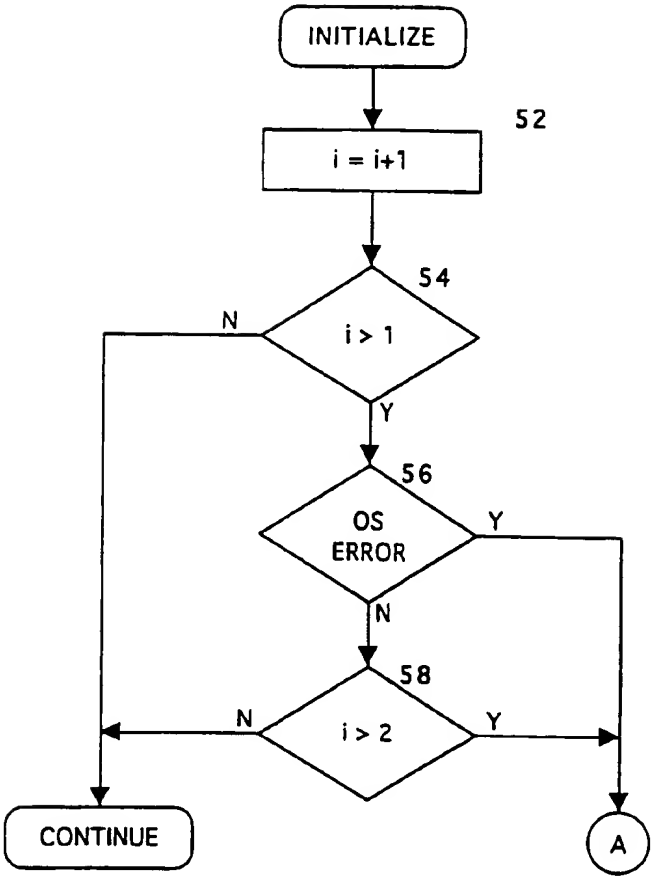


FIG. 8

INTERNATIONAL SEARCH REPORT

Inte. onal Application No
PCT/US 95/01916

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F11/14

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WILLIAM WONG ET AL. 'The MS-DOS Encyclopedia' 1988 , MICROSOFT PRESS , REDMOND, WASHINGTON see page 69, line 23 - page 70, line 14 ---	1-6,8, 10-15, 19,20
A	IBM TECHNICAL DISCLOSURE BULLETIN, vol.36, no.10, October 1993, NEW YORK US pages 329 - 330 'Method to Recover from Failed Boot Drive Using Diskettes' ---	1-20
A	IBM TECHNICAL DISCLOSURE BULLETIN, vol.29, no.2, July 1986, NEW YORK US pages 491 - 493 'Logical IPL selection mechanism' -----	2,4,13

☐ Further documents are listed in the continuation of box C.

☐ Patent family members are listed in annex.

* Special categories of cited documents :

'A' document defining the general state of the art which is not considered to be of particular relevance

'E' earlier document but published on or after the international filing date

'L' document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

'O' document referring to an oral disclosure, use, exhibition or other means

'P' document published prior to the international filing date but later than the priority date claimed

'T' later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

'X' document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

'Y' document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

'&' document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
29 June 1995	06.07.95

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax (+31-70) 340-3016	Corremans, G

Form PCT/ISA/210 (second sheet) (July 1992)